

**MATCHING PROCESSING SYSTEM FOR SOFTWARE AT COMBINING**

Patent Number: JP62006327  
Publication date: 1987-01-13  
Inventor(s): TAKAHASHI YOSHIFUMI  
Applicant(s):: FUJITSU LTD  
Requested Patent: ☐ JP62006327  
Application Number: JP19850146103 19850703  
Priority Number(s):  
IPC Classification: G06F9/06 ; G06F9/44  
EC Classification:  
Equivalents: JP1673209C, JP3041848B

---

**Abstract**

---

**PURPOSE:** To check easily the matching between products A and B by writing the arrival of a prescribed time point to a proper information area of the product A when the product A is utilized internally by the product B and at the same time reading the written contents by the product B.  
**CONSTITUTION:** When a product A like a relational data base, etc. is utilized internally by a product B like a compiler, etc., a proper information area SDT is provided in the product A for the product B. Then a macroinstruction is put into the product B to read out the contents of the area SDT. The functions of alpha, beta and gamma are extended to the product A at a time point (1) and a request is given to set information to the area SDT at a time point (4) when the function to support alpha is given to the product A. While the product B gives announcement to users for an applicable state of the function for alpha and at the same time reads the contents of the area SDT by the macroinstruction for check of the matching between both products A and B.

---

Data supplied from the esp@cenet database - I2

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A)

昭62-6327

⑪ Int.Cl.<sup>4</sup>

G 06 F 9/06  
9/44

識別記号

庁内整理番号

7361-5B  
8120-5B

⑬ 公開 昭和62年(1987)1月13日

審査請求 有 発明の数 1 (全5頁)

⑭ 発明の名称 ソフトウェア組合わせ時の整合化処理方式

⑮ 特 願 昭60-146103

⑯ 出 願 昭60(1985)7月3日

⑰ 発 明 者 高 橋 善 文 川崎市中原区上小田中1015番地 富士通株式会社内  
⑱ 出 願 人 富 士 通 株 式 会 社 川崎市中原区上小田中1015番地  
⑲ 代 理 人 弁 理 士 京 谷 四 郎

## 明 細 書

### 1. 発明の名称

ソフトウェア組合わせ時の整合化処理方式

### 2. 特許請求の範囲

プログラム・プロダクトAと、該プログラム・プロダクトAを内部的に使用するプログラム・プロダクトBとの整合性をチェックするソフトウェア組合わせ時の整合化処理方式であって、上記プログラム・プロダクトAの中に上記プログラム・プロダクトBのための固有情報域を設け、上記プログラム・プロダクトBが上記プログラム・プロダクトAの提供機能の内の或るものをサポートできる時期が来たらこの旨を示すサポート・レベルを上記固有情報域に書き込み、上記プログラム・プロダクトBの中に上記固有情報域の内容を読み取るためのマクロ命令を設け、上記プログラム・プロダクトBが上記マクロ命令により上記固有情報域の内容を読み取って両者間の整合性をチェッ

クすることを特徴とするソフトウェア組合わせ時の整合化処理方式。

### 3. 発明の詳細な説明

#### (概要)

プロダクトAと、該プロダクトAを内部的に利用しているプロダクトBとがあるとき、プロダクトAに固有情報域を設け、プロダクトBがプロダクトAの提供機能αをエンハンス出来る時期がきたら上記固有情報域にその旨を示す情報を書き込むと共に、プロダクトBに上記固有情報域の内容を読み出すためのマクロ命令を記入し、プロダクトBの中でプロダクトAの固有情報域をチェックし、整合をとるソフトウェア組合わせ時の整合化処理方式である。

#### (産業上の利用分野)

本発明は、プロダクトAとプロダクトBの整合化処理方式、例えばリレーショナル・データベースとコンパイラの整合化処理方式に関するものである。

## 〔従来技術と問題点〕

第6図はPL/Iコンパイラとリレーショナル・データベースの整合化処理の従来例を説明する図である。第6図においては、バージョン10レベル10のリレーショナル・データベースと、バージョン10レベル20のリレーショナル・データベースとが図示されている。リレーショナル・データベースは、バージョン及びレベルを示す識別子を有している。バージョン10レベル20のリレーショナル・データベースはXルーチンを有しており、バージョン10レベル10のリレーショナル・データベースはXルーチンを有していない。コンパイラがユーザの作成したソースプログラムに基づいてリレーショナル・データベースのXルーチンを呼び出すオブジェクト命令列を作成する場合、対象となっているリレーショナル・データベースがXルーチンを持っているか否かを調べる必要がある。このため、コンパイラは、第7図に示すように、リレーショナル・データベースが新しいか否か（バージョン10レベル20のものか

否か）を調べ、YesであればXルーチンを呼び出すためのオブジェクトを作成し、Noの場合にはXルーチンを呼び出すためのオブジェクトを作成しない。

ところで、プログラムの修正をPTF (Program Temporary Fix) によって行う場合がある。PTFによるプログラム修正ではバージョン及びレベルを示す識別子は変更されないで、リレーショナル・データベース又はコンパイラに対してPTFによるプログラム修正が行われると、両者が整合しなくなり、ユーザ・プログラムがリレーショナル・データベースの新たに付加された機能を利用しようとする、機械語に変換されたユーザ・プログラムがアペンドしたりすることがある。

## 〔発明の目的〕

本発明は、上記の考案に基づくものであって、プログラム・プロダクトAとプログラム・プロダクトBとの整合性を簡単にチェック出来るようになったソフトウェア組合わせ時の整合化処理方式を提供することを目的としている。

## 〔目的を達成するための手段〕

そしてそのため本発明のソフトウェア組合わせ時の整合化処理方式は、プログラム・プロダクトAと、該プログラム・プロダクトAを内部的に使用するプログラム・プロダクトBとの整合性をチェックするソフトウェア組合わせ時の整合化処理方式であって、上記プログラム・プロダクトAの中に上記プログラム・プロダクトBのための固有情報域を設け、上記プログラム・プロダクトBが上記プログラム・プロダクトAの提供機能の内の或るものをサポートできる時期が来たらこの旨を示すサポート・レベルを上記固有情報域に書き込み、上記プログラム・プロダクトBの中に上記固有情報域の内容を読み取るためのマクロ命令を設け、上記プログラム・プロダクトBが上記マクロ命令により上記固有情報域の内容を読み取って両者の整合性をチェックすることを特徴とするものである。

## 〔発明の実施例〕

以下、本発明を実施例を参照しつつ説明する。

要約すると、本発明は下記のような事項を構成要件とするものである。

- (a) プロダクトAの中にプロダクトBのための固有情報域を設け、プロダクトBがプロダクトAの提供機能 $\alpha$ 、 $\beta$ 、 $\gamma$ のうち取り込んでサポートできる時期が来たら固有情報域の該当するビットをオンにする。
- (b) プロダクトBの中でプロダクトAの中の固有情報域をチェックし、整合を取る。

第1図は本発明の概要を説明するための図である。第1図において、SDIはプロダクトBの固有情報域、DLIBは提供ライブラリ (Distribution Library) をそれぞれ示している。プロダクトAは例えばリレーショナル・データベースであり、プロダクトBはコンパイラである。次に第1図について説明する。①の時点でプロダクトAに対して $\alpha$ 、 $\beta$ 、 $\gamma$ の機能拡張する。②の時点でプロダクトBが $\alpha$ をサポートする機能を作り込んだとすると、プロダクトAの固有情報域に情報設定を依頼する(③)。プロダクトBはユーザに対して④の

PTF を実施すればαの機能を使用できる旨をアナウンスをする。

④のPTF が実施されていれば、t。でαの機能を利用することが出来る。ユーザが組合わせの条件を誤れば固有情報域は000のままであり、ユーザ・プログラムはアベンドしない。しかし、プロダクトB（コンパイラ）はユーザ・プログラムがαの機能を使用したいと言って来た時には適切なエラーメッセージを出力する。プロダクトBにαをサポートさせるための提供ライブラリDLIBと、固有情報域SDTのαに対応するビットに1を書き込むためのPTFとを同時に出荷することがサービスを向上させる上で望ましい。なお、第1図では、α、β、γに対してそれぞれ1ビットずつ割当てられているが、2ビット構成とし、プロダクトBがαの機能をサポート出来れば固有情報域SDTに01を書き込み、αとβの機能をサポート出来れば固有情報域に10を書き込み、α、β、γの機能をサポート出来れば固有情報域に11を書き込むようにしてもよい。

域及びFORTRAN に対する固有情報域は、それぞれ12バイト構成であり、その内の4バイトが実際に使用されている。この4バイトの領域は4ビット単位に分割されている。第1バイトの前半の4ビットは運用機能レベル、第1バイトの後半の4ビットはデータベース機能レベルに割当てられている。以下、図示の通りである。

第4図は言語の機能レベル情報の値の例を示す図である。COBOL 及びPL/Iがパブリック・データベース機能をサポートできる場合には対応する固有情報域の運用レベルの欄（第1バイトの前半）に'2' X（2進で0010）が書き込まれ、FORTRAN がパブリック・データベース機能をサポートできる場合には対応する固有情報域の運用レベルの欄（第1バイトの前半）に'1' X（2進で0001）が書き込まれる。例えばリレーショナル・データベースのカラム保護機能も運用レベルに属しており、COBOL 及びPL/Iがパブリック・データベース機能の外にカラム保護機能もサポートできる場合には対応する固有情報域の運用レベル

第2図はユーザ・プログラム、コンパイラ及びリレーショナル・データベースの関係を示す図である。第2図において、1はユーザ・プログラム、2はPL/Iコンパイラ、3はリレーショナル・データベース、SDT1はCOBOL に対する固有情報域、SDT2はPL/Iに対する固有情報域、SDT3はFORTRAN に対する固有情報域をそれぞれ示している。αの機能を使用したいと言う命令が記入されたPL/I言語で書かされたユーザ・プログラム1をコンパイルする時、PL/Iコンパイラ2は、所定のマクロ命令（このマクロ命令については第5図で説明する）を発行し、リレーショナル・データベース3の固有情報域SDT2の内容を読み取り、αの機能をサポート出来る旨の情報が書き込まれているか否かをチェックし、Yes の場合には、αの機能呼び出すためのオブジェクト命令列を作成し、Noの場合にはαの機能呼び出すためのオブジェクト命令列を作成しない。

第3図は固有情報域の構成を示す図である。COBOL に対する固有情報域、PL/Iに対する固有情報

域の欄に'3' X（2進で0011）が書き込まれ、FORTRAN がパブリック・データベース機能の外にカラム保護機能もサポートできる場合には対応する固有情報域の運用レベルの欄に'2' X（2進で0010）が書き込まれる。第1バイトの後半の4ビットないし第3バイトの後半の4ビットの欄については、上述の説明から理解できるものと思われるので説明は省略する。

第5図は固有情報域内の言語情報を取り出すための言語情報取出しマクロの記述形式を示す図である。このマクロ命令の機能は、指定した言語の固有情報を固有情報域SDT から取り出し、返却すると言うものである。次にオペランドについて説明する。

(a) RCONAD (&RCONAD)

ROB（リレーショナル・データベース）の主制御変(RCON)のアドレスを指定する。

&RCONAD:PTR(31).

(b) LANG ( { CO|PL|FO } )

言語の種別を指定する。

CO:COBOL

PL: L/I

FO:FORTRAN77

(c) RTNAD(&amp;RTNAD)

1 2 バイトの情報返却域のアドレスを指定する。

&amp;RTNAD:PTR(31)

〔発明の効果〕

以上の説明から明らかなように、本発明によれば、プロダクトAが自分のサポート・レベルをプロダクトBの中に持つことになり、従来までアベンドが発生していたものが、適切なメッセージ（組合わせ不良という旨の）が出力されるようになり、障害対策が不要になる。

## 4. 図面の簡単な説明

第1図は本発明の概要を説明するための図、第2図はユーザ・プログラム、コンパイラ及びリレーショナル・データベースの関係を示す図、第3図は固有情報域の構成を示す図、第4図は言語の

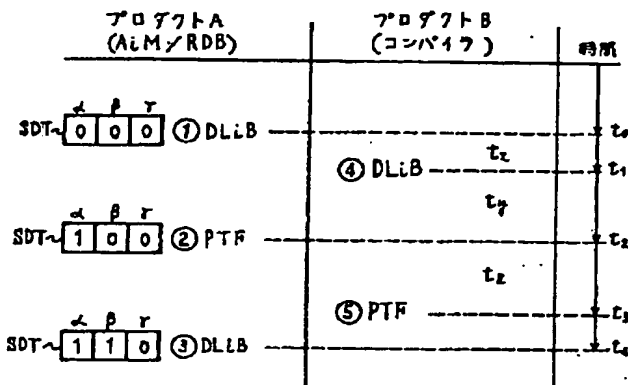
機能レベル情報の値の例を示す図、第5図は言語情報取出しマクロの記述形式を示す図、第6図はPL/Iコンパイラとリレーショナル・データベースの整合化処理の従来例を説明する図、第7図は第6図におけるルーチンXを呼び出すオブジェクト命令列を作成する部分の処理を説明する図である。

1…ユーザ・プログラム、2…PL/Iコンパイラ、3…リレーショナル・データベース、SDT1…COBOLに対する固有情報域、SDT2…PL/Iに対する固有情報域、SDT3…FORTRAN に対する固有情報域。

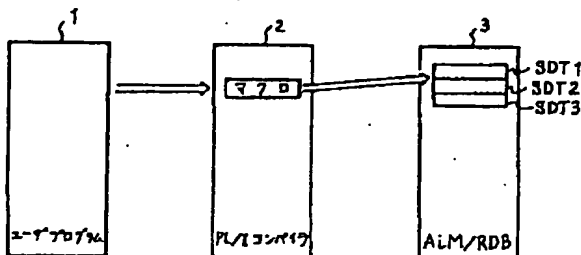
特許出願人 富士通株式会社

代理人弁理士 京谷 四郎

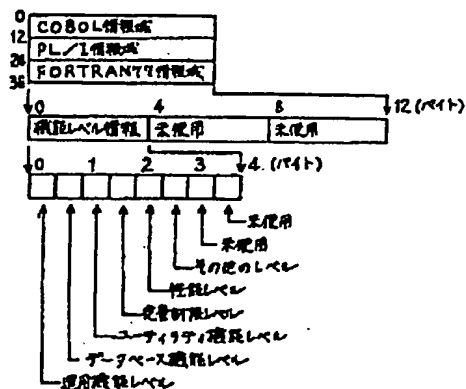
第1図



第2図



第3図



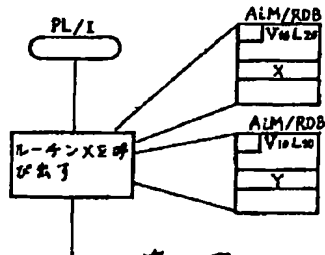
第4図

バイト	COBOL	PL/I	FORTRAN77	'1' Xの意味(注2)
1	運用機能レベル	'2' X	'1' X	バグアップ・データベース機能
2	データベース機能レベル	'1' X	'0' X	集合変数
3	ユーティリティ機能レベル	'1' X	'0' X	実行時コマンド機能
4	変数制限レベル	'1' X	'0' X	VARIABLES/CHARACTER及その拡張
5	性能レベル	'0' X	'0' X	
6	その他のレベル	'1' X	'0' X	制約時8桁メッセージ
7	不使用	'0' X	'0' X	
8	不使用	'0' X	'0' X	

第 5 図

マクロ名	イペランド
JYRMODE	RCOMAD (&RCOMAD) LANG (&COIPLIFO) RTNAD (&RTNAD)

第 6 図



第 7 図

